

Disclaimer:

This English translation is produced by machine translation and may contain errors. The JPO, the INPIT, and those who drafted this document in the original language are not responsible for the result of the translation.

Notes:

1. Untranslatable words are replaced with asterisks (***).
2. Texts in the figures are not translated and shown as it is.

Translated: 05:16:30 JST 05/14/2008

Dictionary: Last updated 04/11/2008 / Priority: 1. Information communication technology (ICT) / 2. Electronic engineering / 3. Chemistry

FULL CONTENTS

[Claim(s)]

[Claim 1] The trace memory which carries out storing maintenance of the predetermined trace information, and the trace memory address for storing said trace information in said trace memory are held. whenever the instruction of a predetermined program was executed, after the trace information concerned was stored in said trace memory -- "1" -- [counter / to increment / trace] The back branching detecting element which compares a value in response to detection directions of a mode indication signal the preceding value of the predetermined program address, and this time, and detects the existence of back branching generating, The branching destination register which latches the value of the branching destination program address at the time of the back branching generating concerned, The branching agency register which latches the value of the branching agency program address at the time of the back branching generating concerned, The loop start register for recognizing the repetition execution portion of said program, In the information control equipment which is equipped with the loop recognition part containing the end register of a loop, and a looped race counter, and stores only the first time and the last round of a repetition execution portion of said program in said trace memory [whether said program execution instruction is an instruction which calls a subroutine / detecting element / to detect / CALL instruction / said program execution instruction] The RET instruction detecting element which detects whether it is the instruction which returns from a subroutine, The call stack which carries out set storing maintenance of two or more each values of the loop start register in said loop recognition part, the end register of a loop, and a looped race counter, respectively, The call stack counter which carries out counting of the number of sets of each value of the loop start register stored in said call stack, the end register of a loop, and a looped race counter, and holds it, Information control equipment characterized by storing only the first time and the last round of a repetition execution portion in said trace memory when preparation ***** is carried out and subroutine

call instruction exists in the repetition execution portion of said program.

[Claim 2] The trace memory which carries out storing maintenance of the predetermined trace information, and the trace memory address for storing said trace information in said trace memory are held. whenever the instruction of a predetermined program was executed, after the trace information concerned was stored in said trace memory -- "1" -- [counter / to increment / trace] The back branching detecting element which compares a value in response to detection directions of a mode indication signal the preceding value of the predetermined program address, and this time, and detects the existence of back branching generating, The branching destination register which latches the value of the branching destination program address at the time of the back branching generating concerned, The branching agency register which latches the value of the branching agency program address at the time of the back branching generating concerned, The loop start register for recognizing the repetition execution portion of said program, In the control method of information control equipment of having the loop recognition part containing the end register of a loop, and a looped race counter, and storing only the first time and the last round of a repetition execution portion of said program in said trace memory The 1st step which initializes said loop start register, the end register of a loop, and a looped race counter, respectively at the time of the start of trace, At the time of the back branching detection by said back branching detecting element after a trace start, the contents of said branching destination register, and the contents of said loop start register, And it sets to the 2nd step which carries out comparative collation of the contents of the said branching former register, and the contents of said end register of a loop, respectively, and said 2nd step. When the contents of said branching destination register, the contents of said loop start register, and the contents of the said branching former register and the contents of said end register of a loop are in agreement In the 3rd step which sets the value of said looped race counter as said trace counter, and said 2nd step When the contents of said branching destination register, the contents of said loop start register, or the contents of the said branching former register and the contents of said end register of a loop are inharmonious The 4th step which sets the value of a said branching former register to said end register of a loop, and sets the value of said trace counter to said loop start register for the value of said branching destination register at said looped race counter, respectively, When the execution of an instruction which calls a subroutine is detected by said CALL instruction detecting element, said loop start register, The 5th step which saves each value of said end register of a loop, and said looped race counter in the position specified with the value of said call stack counter in said call stack, The 6th step to which only "1" makes the value of said call stack counter increase, [with the 7th step which initializes said start register, said end register of a loop, and said RETPU trace counter, respectively, and said RET instruction detecting element] The 8th step to which only "1" decreases the value of said call stack counter when

execution of the instruction which returns from a subroutine is detected, [the value saved in the position specified with the value of said call stack counter in said call stack] The information control method characterized by having the 9th step set as said loop start register, said end register of a loop, and said looped race counter, respectively.

[Claim 3] The trace memory which carries out storing maintenance of the predetermined trace information, and the trace memory address for storing said trace information in said trace memory are held. whenever the instruction of a predetermined program was executed, after the trace information concerned was stored in said trace memory -- "1" -- [counter / to increment / trace] The back branching detecting element which compares a value in response to detection directions of a mode indication signal the preceding value of the predetermined program address, and this time, and detects the existence of back branching generating, The branching destination register which latches the value of the branching destination program address at the time of the back branching generating concerned, The branching agency register which latches the value of the branching agency program address at the time of the back branching generating concerned, The loop start register for recognizing the repetition execution portion of said program, In the control method of information control equipment of having the loop recognition part containing the end register of a loop, and a looped race counter, and storing only the first time and the last round of a repetition execution portion of said program in said trace memory The 1st step which initializes said loop start register, the end register of a loop, and a looped race counter, respectively at the time of the start of trace, At the time of the back branching detection by said back branching detecting element after a trace start, the contents of said branching destination register, and the contents of said loop start register, And it sets to the 2nd step which carries out comparative collation of the contents of the said branching former register, and the contents of said end register of a loop, respectively, and said 2nd step. When both the contents of said branching destination register, the contents of said loop start register, and the contents of the said branching former register and the contents of said end register of a loop are in agreement In the 3rd step which sets the value of said looped race counter as said trace counter, and said 2nd step When the contents of said branching destination register, the contents of said loop start register, or the contents of the said branching former register and the contents of said end register of a loop are inharmonious The 4th step which sets the value of a said branching former register to said end register of a loop, and sets the value of said trace counter to said loop start register for the value of said branching destination register at said looped race counter, respectively, When the execution of an instruction which calls a subroutine is detected by said CALL instruction detecting element, the value of said call stack counter Said loop start register, [with the 5th step to which it is made to increase by total size and said RET instruction detecting element of each size of said end register of a loop, and said looped race counter] When execution of the

instruction which returns from a subroutine is detected, [the value of said call stack counter]
 The information control method characterized by having the 6th step which each size of said loop start register, said end register of a loop, and said looped race counter decreases by total size.

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to information control equipment and its control method.

[0002]

[Description of the Prior Art] The block diagram showing the composition of one example of conventional information control equipment is shown in drawing 3 . This conventional parallel is an example of information control equipment currently indicated by JP,H5-151023,A, and the procedure of trace of the conventional parallel concerned of operation is shown in the flow chart of drawing 4 .

[0003] As shown in drawing 3 , this conventional parallel corresponds to trace information Bath 3 which carries out the mediation transfer of the trace information. While holding the trace memory address for storing in a trace memory 1 the trace memory 1 which stores trace information, and the trace information inputted through trace information Bath 3 The trace counter 4 by which only "1" is incremented with the trace control signal 2 after the trace information concerned is stored in a trace memory 1, The value register 72 is included this time which stores this time value of the preceding value register 71 and the microprogram address 5 which stores the preceding value of the microprogram address 5. When detection directions are carried out by the mode indication signal 6, a value is compared the preceding value of the microprogram address 5, and this time. The back branching detecting element 7 which detects that back branching occurred when a preceding value is beyond a value this time, The branching destination register 8 holding the preceding value register 71 at the time of back branching being detected in the back branching detecting element 7 concerned, With the branching agency register 9 holding this time value register 72 at the time of back branching being detected in the back branching detecting element 7, respectively Loop start register 101-i ($i = 1, 2, \dots, N$), Including end register of loop 102-i, and looped race counter 103-i ($i = 1, 2, \dots, N$), it has the loop recognition part 10-1 holding the information for recognizing the repetition execution portion of a microprogram, 10-2, \dots , 10-N, and is constituted. in addition, [loop start register 101-i in each above-mentioned loop recognition part] It is stored by the head microprogram address of the repetition portion of a microprogram, and [end

register of loop 102-i] The last microprogram address of the repetition portion of a microprogram is stored, and the value of the trace counter 4 at the time of repetition partial recognition of a microprogram is stored in looped race counter 103-i. Moreover, to these loop recognition parts, the loop discernment ID by size relation by which the weighting was carried out is given to each, for example.

[0004] Next, operation of this conventional parallel is explained with reference to drawing 3 and drawing 4 . In addition, the case where only the loop recognition part 10-1, the loop recognition part 10-2, and the three loop recognition part of loop recognition part 10-N the loop recognition part is indicated to be to drawing 3 on account of explanation are prepared is assumed temporarily (in this case). Other loop recognition parts 10-3, 10-4 and 10- (N-1) carry out the outside of the object of explanation of operation once. However, even if it assumes in this way, the generality of explanation of this conventional parallel of operation is not lost.

Explanation of operation shall be given.

[0005] Operation is stopped when operation of this conventional parallel is controlled by the trace control signal 2, operation is started when a trace start is directed by the trace control signal 2 concerned, and a trace halt is directed by the trace control signal 2. The loop start register and the end register of a loop which are contained inside all the loop recognition parts 10-1, 10-2, and 10-N at the time of a trace operation start are initialized in the maximum which can be set as these registers, for example (step ST1). subsequently, [*****] when it is not judged and (step ST2) detected whether back branching was detected in the back branching detecting element 7 while storing trace information in a trace memory 1 through the trace information path 3 -- the trace counter 4 -- "1" -- incrementing (step ST7) -- again -- step ST2 returning -- step ST2 Subsequent operation is started. Moreover, step ST2 When it sets and back branching is detected, it is judged whether the value of the end register 102-1 of a loop of the loop recognition part 10-1 is below a value of the branching destination register 8 or it is beyond the value of the branching agency register 9 (step ST3). step ST3 setting -- affirmation (YES) it is -- ** -- when judged It is judged whether the value of the loop start register 101-1 of the loop recognition part 10-1 is equal to the value of the branching destination register 8, and the value of the end register 102-1 of a loop is equal to the value of the branching agency register 9 (step ST4). Step ST4 when judged with it setting and being denial (NO) The value of the branching destination register 8 is set to the loop start register 101-1 of the loop recognition part 10-1. The value of the branching agency register 9 is set to the end register 102-1 of a loop, the value of the trace counter 4 is set to the looped race counter 103-1, the loop recognition information currently held at the loop recognition part 10-1 is updated (step ST6), and it is said step ST7. It returns. step ST7 as set and mentioned above, while storing trace information in a trace memory 1 through the trace information path 3 -- the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed. said step ST4

[moreover,] setting -- affirmation (YES) it is -- ** -- when judged, the value of the looped race counter 103-1 of the loop recognition part 10-1 is set as the trace counter 4 (step ST5) -- after that -- said step ST7 It returns. step ST7 while setting and storing trace information in a trace memory 1 -- the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed.

[0006] On the other hand, it is step ST3. when judged with it setting and being denial (NO) [be / the value of the end register 102-2 of a loop of the loop recognition part 10-2 / below a value of the branching destination register 8] Or it is judged (step ST10) and it is affirmation (YES) whether it is beyond the value of the branching agency register 9. when judged with it being It is judged whether the value of the loop start register 102-2 of the loop recognition part 10-2 is equal to the value of the branching destination register 8, and the value of the loop start register 102-2 is equal to the value of the branching agency register 9 (step ST11). When judged with denial (NO) in step ST11 The value of the branching destination register 8 is set to the loop start register 101-2 of the loop recognition part 10-2. The value of the branching agency register 9 is set to the end register 103-2 of a loop, the value of the trace counter 4 is set to the looped race counter 103-2, and the loop recognition information currently held at the loop recognition part 10-2 is updated (step ST13). Subsequently, the loop start register 101-1 and the end register 102-1 of a loop of the loop recognition part 10-1 are received. For example, by setting up the maximum which can be set as the register concerned, initialization of the loop recognition part 10-1 is performed (step ST14), and it is said step ST7 after that. It returns. step ST7 while setting and storing trace information in a trace memory 1 -- the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed.

[0007] In addition, in step ST11, it is affirmation (YES). when judged with it being The value of the looped race counter 103-2 of the loop recognition part 10-2 is set to the trace counter 4 (step ST12), and it is said step ST7. while returning and storing trace information in a trace memory 1 the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed. moreover, when judged with it being denial (NO) in said step ST10 [be / the value of end register of loop 102-N of loop recognition part 10-N / below a value of the branching destination register 8] Or it is judged (step ST20) and it is affirmation (YES) whether it is beyond the value of the branching agency register 9. when judged with it being It is judged whether the value of loop start register 101-N of loop recognition part 10-N is equal to the value of the branching destination register 8, and the value of end register of loop 102-N is equal to the value of the branching agency register 9 (step ST21). When judged with it being denial (NO) in step ST21 The value of the branching destination register 8 is set as loop start register 101-[of loop recognition part 10-N] N. The value of the branching agency register 9 is set as end register of loop 102-N, the value of the trace counter 4 is set as looped race counter 103-N, and the loop recognition information currently held at loop recognition part 10-N is

updated (step ST23).

[0008] [as opposed to / the loop recognition part 10-1 the loop start register 101-1 in 10-2 and 101-2, and the end register 102-1 of a loop and 102-2] next, by setting up maximum the loop recognition part 10-1 and initialization of 10-2 are performed (step ST24) -- said step ST7 while returning and storing trace information in a trace memory 1 -- the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed. Moreover, in step ST21, it is affirmation (YES). when judged with it being The value of looped race counter 103-N of loop recognition part 10-N is set as the trace counter 4 (step ST22), and it is said step ST7 similarly. while returning and storing trace information in a trace memory 1 the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed. When judged with it being denial (NO) in step ST20, it is the same, and this is said step ST7. while returning and storing trace information in a trace memory 1 the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed.

[0009]

[Problem to be solved by the invention] In the conventional information control equipment mentioned above and its control method Since collection control of trace is performed by only the information detected in the back branching detecting element When a subroutine call exists in the execution portion which a microprogram repeats and a repetition execution portion exists further in the child routine called there [the microprogram address of the head of the repetition execution portion of the child routine concerned] [address / of the last of the repetition execution portion of the parent routine which called the child routine / more than / microprogram] When [or] the last microprogram address of the repetition execution portion of a child routine agrees on the conditions below of the microprogram address of the last of the repetition execution portion of said parent routine It becomes the situation where the loop recognition information on a parent routine will be canceled from a loop recognition part at the same time the loop recognition information on a child routine is registered into a loop recognition part, when back branching occurs in said child routine.

[0010] As a result, although the repetition execution portion of said child routine is traced only in the first time and the last round, all for a repeat count will be traced about the repetition execution portion of said parent routine. Thus, in conventional information control equipment and its conventional control method, when a subroutine call exists in the repetition execution portion of a microprogram, there is a fault that efficient trace information is uncollectible.

[0011] The purpose of this invention solves the above-mentioned technical problem, recognizes the subroutine control section in a my KUROPU gram by detection of subroutine call instruction and the return instruction from a subroutine, and there is in realizing the information control equipment which enables collection of efficient trace information, and its control method.

[0012]

[Means for solving problem] The trace memory to which the information control equipment of the 1st invention carries out storing maintenance of the predetermined trace information, The trace memory address for storing said trace information in said trace memory is held. whenever the instruction of a predetermined program was executed, after the trace information concerned was stored in said trace memory -- "1" -- [counter / to increment / trace] The back branching detecting element which compares a value in response to detection directions of a mode indication signal the preceding value of the predetermined program address, and this time, and detects the existence of back branching generating, The branching destination register which latches the value of the branching destination program address at the time of the back branching generating concerned, The branching agency register which latches the value of the branching agency program address at the time of the back branching generating concerned, The loop start register for recognizing the repetition execution portion of said program, In the information control equipment which is equipped with the loop recognition part containing the end register of a loop, and a looped race counter, and stores only the first time and the last round of a repetition execution portion of said program in said trace memory [whether said program execution instruction is an instruction which calls a subroutine / detecting element / to detect / CALL instruction / said program execution instruction] The RET instruction detecting element which detects whether it is the instruction which returns from a subroutine, The call stack which carries out set storing maintenance of two or more each values of the loop start register in said loop recognition part, the end register of a loop, and a looped race counter, respectively, The call stack counter which carries out counting of the number of sets of each value of the loop start register stored in said call stack, the end register of a loop, and a looped race counter, and holds it, When preparation ***** is carried out and subroutine call instruction exists in the repetition execution portion of said program, it is characterized by storing only the first time and the last round of a repetition execution portion in said trace memory.

[0013] Moreover, the trace memory to which the information control method of the 2nd invention carries out storing maintenance of the predetermined trace information, The trace memory address for storing said trace information in said trace memory is held. whenever the instruction of a predetermined program was executed, after the trace information concerned was stored in said trace memory -- "1" -- [counter / to increment / trace] The back branching detecting element which compares a value in response to detection directions of a mode indication signal the preceding value of the predetermined program address, and this time, and detects the existence of back branching generating, The branching destination register which latches the value of the branching destination program address at the time of the back branching generating concerned, The branching agency register which latches the value of the

branching agency program address at the time of the back branching generating concerned, The loop start register for recognizing the repetition execution portion of said program, In the control method of information control equipment of having the loop recognition part containing the end register of a loop, and a looped race counter, and storing only the first time and the last round of a repetition execution portion of said program in said trace memory The 1st step which initializes said loop start register, the end register of a loop, and a looped race counter, respectively at the time of the start of trace, At the time of the back branching detection by said back branching detecting element after a trace start, the contents of said branching destination register, and the contents of said loop start register, And it sets to the 2nd step which carries out comparative collation of the contents of the said branching former register, and the contents of said end register of a loop, respectively, and said 2nd step. When both the contents of said branching destination register, the contents of said loop start register, and the contents of the said branching former register and the contents of said end register of a loop are in agreement In the 3rd step which sets the value of said looped race counter as said trace counter, and said 2nd step When the contents of said branching destination register, the contents of said loop start register, or the contents of the said branching former register and the contents of said end register of a loop are inharmonious The 4th step which sets the value of a said branching former register to said end register of a loop, and sets the value of said trace counter to said loop start register for the value of said branching destination register at said looped race counter, respectively, When the execution of an instruction which calls a subroutine is detected by said CALL instruction detecting element, said loop start register, The 5th step which saves each value of said end register of a loop, and said looped race counter in the position specified with the value of said call stack counter in said call stack, The 6th step to which only "1" makes the value of said call stack counter increase, [with the 7th step which initializes said start register, said end register of a loop, and said REPU trace counter, respectively, and said RET instruction detecting element] The 8th step to which only "1" decreases the value of said call stack counter when execution of the instruction which returns from a subroutine is detected, It is characterized by having the 9th step which sets the value saved in the position specified with the value of said call stack counter in said call stack, respectively as said loop start register, said end register of a loop, and said looped race counter.

[0014] Furthermore, the trace memory to which the information control method of the 3rd invention carries out storing maintenance of the predetermined trace information, The trace memory address for storing said trace information in said trace memory is held. whenever the instruction of a predetermined program was executed, after the trace information concerned was stored in said trace memory -- "1" -- [counter / to increment / trace] The back branching detecting element which compares a value in response to detection directions of a mode

indication signal the preceding value of the predetermined program address, and this time, and detects the existence of back branching generating, The branching destination register which latches the value of the branching destination program address at the time of the back branching generating concerned, The branching agency register which latches the value of the branching agency program address at the time of the back branching generating concerned, The loop start register for recognizing the repetition execution portion of said program, In the control method of information control equipment of having the loop recognition part containing the end register of a loop, and a looped race counter, and storing only the first time and the last round of a repetition execution portion of said program in said trace memory The 1st step which initializes said loop start register, the end register of a loop, and a looped race counter, respectively at the time of the start of trace, At the time of the back branching detection by said back branching detecting element after a trace start, the contents of said branching destination register, and the contents of said loop start register, And it sets to the 2nd step which carries out comparative collation of the contents of the said branching former register, and the contents of said end register of a loop, respectively, and said 2nd step. When both the contents of said branching destination register, the contents of said loop start register, and the contents of the said branching former register and the contents of said end register of a loop are in agreement In the 3rd step which sets the value of said looped race counter as said trace counter, and said 2nd step When the contents of said branching destination register, the contents of said loop start register, or the contents of the said branching former register and the contents of said end register of a loop are inharmonious The 4th step which sets the value of a said branching former register to said end register of a loop, and sets the value of said trace counter to said loop start register for the value of said branching destination register at said looped race counter, respectively, When the execution of an instruction which calls a subroutine is detected by said CALL instruction detecting element, the value of said call stack counter Said loop start register, [with the 5th step to which it is made to increase by total size and said RET instruction detecting element of each size of said end register of a loop, and said looped race counter] When execution of the instruction which returns from a subroutine is detected, [the value of said call stack counter] It is characterized by having the 6th step which each size of said loop start register, said end register of a loop, and said looped race counter decreases by total size.

[0015]

[Mode for carrying out the invention] Next, this invention is explained with reference to Drawings.

[0016] Drawing 1 is the block diagram showing the composition of one embodiment of this invention. As shown in drawing 1 , this embodiment corresponds to trace information Bath 3 which carries out the mediation transfer of the trace information. While holding the trace

memory address for storing in a trace memory 1 the trace memory 1 which stores trace information, and the trace information inputted through trace information Bath 3 The trace counter 4 by which only "1" is incremented with the trace control signal 2 after the trace information concerned is stored in a trace memory 1, The value register 72 is included this time which stores this time value of the preceding value register 71 and the microprogram address 5 which stores the preceding value of the microprogram address 5. When detection directions are carried out by the mode indication signal 6, a value is compared the preceding value of the microprogram address 5, and this time. The back branching detecting element 7 which detects that back branching occurred when a preceding value is beyond a value this time, The branching destination register 8 holding the preceding value register 71 at the time of back branching being detected in the back branching detecting element 7 concerned, With the branching agency register 9 holding this time value register 72 at the time of back branching being detected in the back branching detecting element 7, respectively Loop start register 101-i ($i = 1, 2, \dots, N$), End register of loop 102-i and looped race counter 103-i ($i = 1, 2, \dots, N$) are included. The loop recognition part 10-1 holding the information for recognizing the repetition execution portion of a microprogram, 10-2, \dots , 10-N, The CALL instruction detecting element 21 which detects whether the run command of a microprogram is an instruction which calls a subroutine, The RET instruction detecting element 22 which detects whether the run command of a microprogram is an instruction which returns from a subroutine, At the time of a subroutine call, the loop recognition part 10-1, 10-2, \dots , The call stack 23 which is the FILO type field used in order to take out the loop recognition information which the loop recognition information currently held at 10-N is saved, and is saved at these loop recognition parts at the time of the return from a subroutine, It has the call stack counter 24 in which the number of the loop recognition information stored in the call stack 23 concerned is shown, and is constituted.

[0017] In addition, like the case of conventional parallel [loop start register 101-i in each above-mentioned loop recognition part] It is stored by the head microprogram address of the repetition portion of a microprogram, and [end register of loop 102-i] The last microprogram address of the repetition portion of a microprogram is stored, and the value of the trace counter 4 at the time of repetition partial recognition of a microprogram is stored in looped race counter 103-i. Moreover, to these loop recognition parts, the loop discernment ID by size relation by which the weighting was carried out is given to each, for example.

[0018] Moreover, drawing 2 is a flow chart which shows the procedure of this embodiment of operation. By contrast with drawing 4 , it sets to this embodiment so that clearly. The CALL instruction detecting element 21, the RET instruction detecting element 22, the call stack 23, and call stack counter 24 grade with addition of the equipment configuration to include as a processing step The step containing step ST31, ST32, ST33, ST34, ST35, and ST36 grade is

newly added.

[0019] Next, operation of this conventional parallel is explained with reference to drawing 1 and drawing 2 . In addition, explanation of operation shall be given supposing the case where only the loop recognition part 10-1, the loop recognition part 10-2, and the three loop recognition part of loop recognition part 10-N the loop recognition part is indicated to be to drawing 1 on account of explanation are prepared like the case of conventional parallel.

[0020] The loop start register and the end register of a loop which are contained inside all the loop recognition parts at the time of a trace operation start are initialized (step ST1).

Subsequently, the call stack counter 24 is initialized by "0" (step ST31). It is judged whether during trace operation, it was detected in the CALL instruction detecting element 21 that subroutine call instruction was executed (step ST32). When execution of subroutine call instruction is detected The value of the loop start register 101-1 currently held at the loop recognition part 10-1, 10-2, and 10-N, 101-2, and 101-N, The end register 102-1 of a loop, 102-2, and the value of 102 Ns, [loop recognition information including the value of the looped race counter 103-1, 103-2, and 103-N] being saved in the predetermined position in the call stack 23 specified with the value of the call stack counter 24 -- the call stack counter 24 -- "1" -- it increments (step ST33). And while the loop recognition information in the loop recognition part 10-1, 10-2, and 10-N is initialized (step ST34) and storing trace information in a trace memory 1 through trace information Bath 3 the trace counter 4 -- "1" -- incrementing (step ST7) -- again -- step ST2 returning -- step ST2 Subsequent operation is started. moreover, when execution of subroutine call instruction is not detected in step ST32 It is judged whether during trace operation, it was detected in the RET instruction detecting element 22 that subroutine return instruction was executed (step ST35). When execution of subroutine return instruction is detected the call stack counter 24 -- "1" -- [a decrement is carried out, and the value saved in the predetermined position in the call stack 23 specified by the call stack counter 24 is taken out, and] It is set up as loop recognition information in the loop recognition part 10-1, 10-2, and 10-N (step ST36). moreover, in step ST35 when execution of subroutine return instruction is not detected [*****] when it is not judged and (step ST2) detected whether back branching was detected in the back branching detecting element 7 while storing trace information in a trace memory 1 through trace information Bath 3 -- the trace counter 4 -- "1" -- incrementing (step ST7) -- again -- step ST2 returning -- step ST2 Subsequent operation is started. Moreover, step ST2 When it sets and back branching is detected, it is judged whether the value of the end register 102-1 of a loop of the loop recognition part 10-1 is below a value of the branching destination register 8 or it is beyond the value of the branching agency register 9 (step ST3).

[0021] step ST3 setting -- affirmation (YES) it is -- ** -- when judged It is judged whether the value of the loop start register 101-1 of the loop recognition part 10-1 is equal to the value of

the branching destination register 8, and the value of the end register 102-1 of a loop is equal to the value of the branching agency register 9 (step ST4). Step ST4 when judged with it setting and being denial (NO) The value of the branching destination register 8 is set to the loop start register 101-1 of the loop recognition part 10-1. The value of the branching agency register 9 is set to the end register 102-1 of a loop, the value of the trace counter 4 is set to the looped race counter 103-1, the loop recognition information currently held at the loop recognition part 10-1 is updated (step ST6), and it is said step ST7. It returns. step ST7 as set and mentioned above, while storing trace information in a trace memory 1 through the trace information path 3 -- the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed. said step ST4 [moreover,] setting -- affirmation (YES) it is -- ** -- when judged, the value of the looped race counter 103-1 of the loop recognition part 10-1 is set as the trace counter 4 (step ST5) -- after that -- said step ST7 It returns. while storing trace information in a trace memory 1 in step ST7 -- the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed.

[0022] On the other hand, it is step ST3. when judged with it setting and being denial (NO) [be / the value of the end register 102-2 of a loop of the loop recognition part 10-2 / below a value of the branching destination register 8] Or it is judged (step ST10) and it is affirmation (YES) whether it is beyond the value of the branching agency register 9. when judged with it being It is judged whether the value of the loop start register 102-2 of the loop recognition part 10-2 is equal to the value of the branching destination register 8, and the value of the loop start register 102-2 is equal to the value of the branching agency register 9 (step ST11). When judged with denial (NO) in step ST11 The value of the branching destination register 8 is set to the loop start register 101-2 of the loop recognition part 10-2. The value of the branching agency register 9 is set to the end register 103-2 of a loop, the value of the trace counter 4 is set to the looped race counter 103-2, and the loop recognition information currently held at the loop recognition part 10-2 is updated (step ST13). Subsequently, the loop start register 101-1 and the end register 102-1 of a loop of the loop recognition part 10-1 are received. For example, by setting up the maximum which can be set as the register concerned, initialization of the loop recognition part 10-1 is performed (step ST14), and it is said step ST7 after that. It returns. step ST7 while setting and storing trace information in a trace memory 1 -- the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed.

[0023] In addition, in step ST11, it is affirmation (YES). when judged with it being The value of the looped race counter 103-2 of the loop recognition part 10-2 is set to the trace counter 4 (step ST12), and it is said step ST7. while returning and storing trace information in a trace memory 1 the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed. moreover, when judged with it being denial (NO) in said step ST10 [be / the value of end register of loop 102-N of loop recognition part 10-N / below a value of

the branching destination register 8] Or it is judged (step ST20) and it is affirmation (YES) whether it is beyond the value of the branching agency register 9. when judged with it being It is judged whether the value of loop start register 101-N of loop recognition part 10-N is equal to the value of the branching destination register 8, and the value of end register of loop 102-N is equal to the value of the branching agency register 9 (step ST21). When judged with it being denial (NO) in step ST21 The value of the branching destination register 8 is set as loop start register 101-[of loop recognition part 10-N] N. The value of the branching agency register 9 is set as end register of loop 102-N, the value of the trace counter 4 is set as looped race counter 103-N, and the loop recognition information currently held at loop recognition part 10-N is updated (step ST23).

[0024] [as opposed to / the loop recognition part 10-1 the loop start register 101-1 in 10-2 and 101-2, and the end register 102-1 of a loop and 102-2] next, by setting up maximum the loop recognition part 10-1 and initialization of 10-2 are performed (step ST24) -- said step ST7 while returning and storing trace information in a trace memory 1 -- the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed. Moreover, in step ST21, it is affirmation (YES). when judged with it being The value of looped race counter 103-N of loop recognition part 10-N is set as the trace counter 4 (step ST22), and it is said step ST7 similarly. while returning and storing trace information in a trace memory 1 the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed. When judged with it being denial (NO) in step ST20, it is the same, and this is said step ST7. while returning and storing trace information in a trace memory 1 the trace counter 4 -- "1" -- incrementing -- again -- step ST2 Subsequent processings are performed.

[0025] Since the loop recognition information currently held in the loop recognition part can be changed in between the parent routine which called the subroutine as mentioned above, and the called child routines Also in the microprogram containing a subroutine, it becomes possible to perform efficient trace operation. That is, according to this invention, the subroutine control section in a my KUOPU gram is recognized by detection of subroutine call instruction and the return instruction from a subroutine, and the information control equipment which enables collection of efficient trace information, and its control method are realized.

[0026] In addition, the loop start register with which the call stack counter is stored in the call stack in the above-mentioned explanation, Carry out counting of the number of sets of each value of the end register of a loop, and a looped race counter, and it is held. When it is detected in a CALL instruction detecting element during trace operation that subroutine call instruction was executed After the loop recognition information currently held at all the loop recognition parts is stored in a call stack a call stack counter -- "1" -- incrementing -- moreover, trace -- [a thing] working when it is detected in the RET instruction detecting-element section that subroutine return instruction was executed [after "1" decrement's carrying out a call stack

counter, explained the information control method of having taken out the loop recognition information stored in the call stack, and setting it as all the loop recognition parts, but] When the call stack is mounted by memory, the call stack counter holds the address of said call stack apart from this control method and it is detected in a CALL instruction detecting element during trace operation that subroutine call instruction was executed After storing in a call stack the loop recognition information currently held in all the loop recognition parts Each size of a loop start register, the end register of a loop, and a looped race counter makes the value of a call stack counter increase by sum total Sais. moreover, when it is detected in an RET instruction detecting element during trace operation that subroutine return instruction was executed The loop recognition information stored in the call stack in the value of the call stack counter after each size of a loop start register, the end register of a loop, and a looped race counter decreases by sum total Sais is taken out. You may use the information control method of setting it as all the loop recognition parts. Moreover, with reference to one example of application of a microprogram, it is explained below concretely what kind of thing the effect of this invention is.

[0027] Drawing 5 (a) is one example of the microprogram having contained the subroutine. Moreover, drawing 5 (b) is the figure showing the result of having traced the microprogram shown in above-mentioned drawing 5 (a) with conventional information control equipment, and drawing 5 (c) is the figure showing the result of having traced the microprogram shown in drawing 5 (a) by this invention. In conventional information control equipment and its conventional control method As shown in drawing 5 (b), [time / of repetitive operation of processing / of parent routine / A, and Processing B / the / 2nd / trace information / of eye time /, and trace information of first time / of repetitive operation of the processing C of routine at that time /, and the last round] (N-1) To remaining in a trace memory, in this invention, as shown in drawing 5 (c), it is improved so that said trace information may not remain in a trace memory. [the trace memory consumption in Processing A Processing B, and Processing C of MicroPro GURAKU which are shown in drawing 5 (a)] if both 500 bytes, Repetitions N [several], and M of both are assumed to be 10 To the consumption of the trace memory in conventional information control equipment and its method being 16000 bytes, the consumption of the trace memory in this invention stops at 4000 bytes, and the use efficiency of a trace memory is improved 4 times. Therefore, by the trace memory of the same capacity, it can leave the execution result of a 4-time-as many microprogram as this in the trace memory concerned, and the range which can debug the microprogram by trace data is expanded by 4 times, and can raise debugging efficiency.

[0028]

[Effect of the Invention] [this invention] by having a CALL instruction detecting element and an RET instruction detecting element as explained above Detect execution of the subroutine call

instruction of a microprogram, and execution of return instruction, and it sets at the time of detection of the subroutine instruction concerned. Hold the loop recognition information on a loop recognition part that the detection time was used, to the call stack, and it sets at the time of subroutine return instruction detection. As the loop recognition information currently held by the subroutine call last time at the call stack is returned to a loop recognition part, it sets between a parent routine and a child routine. By changing the loop recognition information currently held at the loop recognition part, and having been made to perform trace operation [when the subroutine call is contained in the repetition execution portion of a microprogram] In trace of the microprogram which trace information will be left behind only to the first time and the last round of a repetition execution portion, and contained the subroutine The trace memory of the same capacity enables it to leave the execution result of a much more wide range microprogram to a trace memory, and the range of the microprogram by trace data which can be debugged is expanded by this. It is effective in the ability to raise debugging efficiency.

[Brief Description of the Drawings]

[Drawing 1] It is the block diagram showing the composition of one embodiment of this invention.

[Drawing 2] It is the figure showing the flow chart of the procedure of operation in said one embodiment.

[Drawing 3] It is the block diagram showing the composition of conventional parallel.

[Drawing 4] It is the figure showing the flow chart of the procedure of operation in said conventional parallel.

[Drawing 5] It is the figure showing the trace data of a microprogram and a microprogram.

[Explanations of letters or numerals]

1 Trace Counter

2 Trace Control Signal

3 Trace Information Bath

4 Trace Counter

5 Microprogram Address

6 Mode Indication Signal

7 Back Branching Detecting Element

8 Branching Destination Register

9 Branching Agency Register

10-1, 10-2, ..., 10-N Loop recognition part

21 CALL Instruction Detecting Element

22 RET Instruction Detecting Element

23 Call Stack

24 Call Stack Counter

101-1, 101-2,, 101-N Loop start register

102-1, 102-2,, 102-N End register of a loop

103-1, 103-2,, 103-N Looped race counter

ST1 - ST24, ST31-ST36 Step

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平9-128264

(43)公開日 平成9年(1997)5月16日

(51)Int.Cl.⁶

G 0 6 F 11/28

識別記号

3 1 0

庁内整理番号

7313-5B

F I

G 0 6 F 11/28

技術表示箇所

3 1 0 B

審査請求 有 請求項の数 3 O L (全 14 頁)

(21)出願番号 特願平7-282963

(22)出願日 平成7年(1995)10月31日

(71)出願人 000232036

日本電気アイシーマイコンシステム株式会社
神奈川県川崎市中原区小杉町1丁目403番
53

(72)発明者 酒井 貴

神奈川県川崎市中原区小杉町一丁目403番
53 日本電気アイシーマイコンシステム株
式会社内

(72)発明者 橋本 一也

神奈川県川崎市中原区小杉町一丁目403番
53 日本電気アイシーマイコンシステム株
式会社内

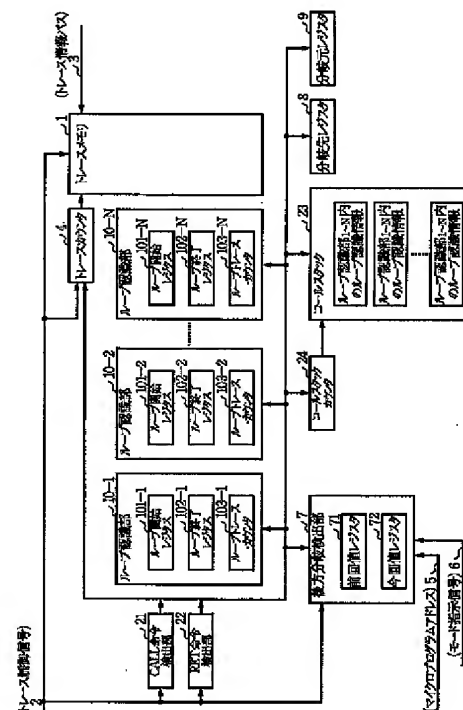
(74)代理人 弁理士 京本 直樹 (外2名)

(54)【発明の名称】 情報制御装置及びその制御方法

(57)【要約】

【課題】情報制御装置及びその制御方法によるトレース収集効率を改善する。

【解決手段】トレース情報を格納するトレースメモリ1と、トレースメモリ・アドレスを保持し、トレース情報格納後に“1”インクリメントするトレースカウンタ4と、マイクロプログラム・アドレス5の前回値と今回値とを比較して後方分岐の発生を検出する後方分岐検出部7と、後方分岐検出時に、それぞれ前回値レジスタ71および今回値レジスタ72を保持する分岐先レジスタ8および分岐元レジスタ9と、マイクロプログラムの繰返し実行部分の認識情報を保持するループ認識部10-1〜10-Nと、マイクロプログラムの実行命令がサブルーチンと呼出す命令かを検出するCALL命令検出部21と、マイクロプログラムの実行命令がサブルーチンからの復帰命令かを検出するRET命令検出部22と、コールスタック23と、コールスタック・カウンタ24とを備えて構成される。



【特許請求の範囲】

【請求項1】 所定のトレース情報を格納保持するトレースメモリと、前記トレース情報を前記トレースメモリに格納するためのトレースメモリ・アドレスを保持し、所定のプログラムの命令が実行される都度、前記トレースメモリに当該トレース情報が格納された後に“1”インクリメントするトレースカウンタと、モード指示信号の検出指示を受けて所定のプログラム・アドレスの前回値と今回値とを比較して後方分岐発生の有無を検出する後方分岐検出部と、当該後方分岐発生時における分岐先プログラム・アドレスの値をラッチする分岐先レジスタと、当該後方分岐発生時における分岐元プログラム・アドレスの値をラッチする分岐元レジスタと、前記プログラムの繰返し実行部分を認識するためのループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタを含むループ認識部とを備え、前記プログラムの繰返し実行部分の初回と最終回のみを前記トレースメモリに格納する情報制御装置において、前記プログラムの実行命令が、サブルーチンを呼出す命令であるか否かを検出するCALL命令検出部と、前記プログラムの実行命令が、サブルーチンから復帰する命令であるか否かを検出するRET命令検出部と、前記ループ認識部内のループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタのそれぞれの値を、それぞれ複数セット格納保持するコールスタックと、前記コールスタックに格納されているループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタのそれぞれの値のセット数を計数して保持するコールスタック・カウンタと、を備えて構成され、前記プログラムの繰返し実行部分にサブルーチン呼出し命令が存在する場合に、繰返し実行部分の初回と最終回のみを前記トレースメモリ内に格納することを特徴とする情報制御装置。

【請求項2】 所定のトレース情報を格納保持するトレースメモリと、前記トレース情報を前記トレースメモリに格納するためのトレースメモリ・アドレスを保持し、所定のプログラムの命令が実行される都度、前記トレースメモリに当該トレース情報が格納された後に“1”インクリメントするトレースカウンタと、モード指示信号の検出指示を受けて所定のプログラム・アドレスの前回値と今回値とを比較して後方分岐発生の有無を検出する後方分岐検出部と、当該後方分岐発生時における分岐先プログラム・アドレスの値をラッチする分岐先レジスタと、当該後方分岐発生時における分岐元プログラム・アドレスの値をラッチする分岐元レジスタと、前記プログラムの繰返し実行部分を認識するためのループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタを含むループ認識部とを備え、前記プログラムの繰返し実行部分の初回と最終回のみを前記トレースメモリ

に格納する情報制御装置の制御方法において、トレースの開始時に、前記ループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタをそれぞれ初期化する第1のステップと、トレース開始後の前記後方分岐検出部による後方分岐検出時に、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、ならびに前記分岐元レジスタの内容と前記ループ終了レジスタの内容とを、それぞれ比較照合する第2のステップと、前記第2のステップにおいて、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、ならびに前記分岐元レジスタの内容と前記ループ終了レジスタの内容とが一致した場合に、前記トレース・カウンタに前記ループトレース・カウンタの値を設定する第3のステップと、前記第2のステップにおいて、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、或は前記分岐元レジスタの内容と前記ループ終了レジスタの内容とが不一致の場合に、前記ループ開始レジスタには前記分岐先レジスタの値を、前記ループ終了レジスタには前記分岐元レジスタの値を、前記ループトレース・カウンタには前記トレース・カウンタの値をそれぞれ設定する第4のステップと、前記CALL命令検出部により、サブルーチンを呼出す命令の実行が検出された時に、前記ループ開始レジスタ、前記ループ終了レジスタおよび前記ループトレース・カウンタのそれぞれの値を、前記コールスタック内の前記コールスタック・カウンタの値により指定される位置に保存する第5のステップと、前記コールスタック・カウンタの値を“1”だけ増加させる第6のステップと、前記開始レジスタ、前記ループ終了レジスタおよび前記ループトレース・カウンタをそれぞれ初期化する第7のステップと、前記RET命令検出部により、サブルーチンから復帰する命令の実行が検出された時に、前記コールスタック・カウンタの値を“1”だけ減少させる第8のステップと、前記コールスタック内の前記コールスタック・カウンタの値により指定される位置に保存されている値を、前記ループ開始レジスタ、前記ループ終了レジスタおよび前記ループトレース・カウンタにそれぞれ設定する第9のステップと、を有することを特徴とする情報制御方法。

【請求項3】 所定のトレース情報を格納保持するトレースメモリと、前記トレース情報を前記トレースメモリに格納するためのトレースメモリ・アドレスを保持し、所定のプログラムの命令が実行される都度、前記トレースメモリに当該トレース情報が格納された後に“1”インクリメントするトレースカウンタと、モード指示信号の検出指示を受けて所定のプログラム・アドレスの前回

値と今回値とを比較して後方分岐発生の有無を検出する後方分岐検出部と、当該後方分岐発生時における分岐先プログラム・アドレスの値をラッチする分岐先レジスタと、当該後方分岐発生時における分岐元プログラム・アドレスの値をラッチする分岐元レジスタと、前記プログラムの繰返し実行部分を認識するためのループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタを含むループ認識部とを備え、前記プログラムの繰返し実行部分の初回と最終回のみを前記トレースメモリに格納する情報制御装置の制御方法において、

トレースの開始時に、前記ループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタをそれぞれ初期化する第1のステップと、

トレース開始後の前記後方分岐検出部による後方分岐検出時に、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、ならびに前記分岐元レジスタの内容と前記ループ終了レジスタの内容とを、それぞれ比較照合する第2のステップと、

前記第2のステップにおいて、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、ならびに前記分岐元レジスタの内容と前記ループ終了レジスタの内容とが共に一致した場合に、前記トレース・カウンタに前記ループトレース・カウンタの値を設定する第3のステップと、

前記第2のステップにおいて、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、或は前記分岐元レジスタの内容と前記ループ終了レジスタの内容とが不一致の場合に、前記ループ開始レジスタには前記分岐先レジスタの値を、前記ループ終了レジスタには前記分岐元レジスタの値を、前記ループトレース・カウンタには前記

トレース・カウンタの値をそれぞれ設定する第4のステップと、
前記CALL命令検出部により、サブルーチンを呼出す命令の実行が検出された時に、前記コールスタック・カウンタの値を、前記ループ開始レジスタ、前記ループ終了レジスタおよび前記ループトレース・カウンタのそれぞれのサイズの合計サイズ分増加させる第5のステップと、

前記RET命令検出部により、サブルーチンから復帰する命令の実行が検出された時に、前記コールスタック・カウンタの値を、前記ループ開始レジスタ、前記ループ終了レジスタおよび前記ループトレース・カウンタのそれぞれのサイズの合計サイズ分減少させる第6のステップと、

を有することを特徴とする情報制御方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は情報制御装置及びその制御方法に関する。

【0002】

【従来の技術】従来の情報制御装置の1例の構成を示すブロック図が図3に示される。本従来例は、特開平5-151023号公報に開示されている情報制御装置例であり、当該従来例のトレースの動作手順は、図4のフローチャートに示されている。

【0003】図3に示されるように、本従来例は、トレース情報を媒介伝達するトレース情報バス3に対応して、トレース情報を格納するトレースメモリ1と、トレース情報バス3を介して入力されるトレース情報を、トレースメモリ1に格納するためのトレースメモリ・アドレスを保持するとともに、当該トレース情報がトレースメモリ1に格納された後に、トレース制御信号2により“1”だけインクリメントされるトレースカウンタ4と、マイクロプログラム・アドレス5の前回値を格納する前回値レジスタ71およびマイクロプログラム・アドレス5の今回値を格納する今回値レジスタ72を含み、モード指示信号6により検出指示される場合に、マイクロプログラム・アドレス5の前回値と今回値とを比較して、前回値が今回値以上である場合に、後方分岐が発生したことを検出する後方分岐検出部7と、当該後方分岐検出部7において後方分岐が検出された際の前回値レジスタ71を保持する分岐先レジスタ8と、後方分岐検出部7において後方分岐が検出された際の今回値レジスタ72を保持する分岐元レジスタ9と、それぞれループ開始レジスタ101-i (i=1, 2, ……………, N)、ループ終了レジスタ102-i、およびループトレース・カウンタ103-i (i=1, 2, ……………, N)を含み、マイクロプログラムの繰返し実行部分を認識するための情報を保持するループ認識部10-1、10-2、……………、10-Nとを備えて構成される。なお、上記の各ループ認識部内のループ開始レジスタ101-iには、マイクロプログラムの繰返し部分の先頭マイクロプログラム・アドレスが格納され、ループ終了レジスタ102-iには、マイクロプログラムの繰返し部分の最終マイクロプログラム・アドレスが格納され、ループトレース・カウンタ103-iには、マイクロプログラムの繰返し部分認識時のトレースカウンタ4の値が格納される。また、これらのループ認識部に対しては、それぞれに、例えば大小関係による重み付けされたループ識別IDが付与されている。

【0004】次に、図3および図4を参照して、本従来例の動作について説明する。なお、説明の都合上、ループ認識部が、図3に記載されているループ認識部10-1、ループ認識部10-2およびループ認識部10-Nの3個のループ認識部のみが設けられている場合を仮に想定して（この場合、他のループ認識部10-3、10-4、……………、10-(N-1)は、一応動作説明の対象外とする。しかし、このように想定しても、本従来例の動作説明の一般性は失われない。）、動作説明を行うものとする。

【0005】本従来例の動作は、トレース制御信号2により制御されており、当該トレース制御信号2によりトレース開始が指示される時点において動作が開始され、またトレース制御信号2によりトレース停止が指示される時点において動作が停止される。トレース動作開始時においては、全てのループ認識部10-1、10-2および10-Nの内部に含まれるループ開始レジスタおよびループ終了レジスタは、例えば、これらのレジスタに設定可能な最大値において初期化される(ステップST1)。次いで、後方分岐検出部7においては後方分岐が検出されたか否かが判定されて(ステップST2)、検出されていない場合には、トレース情報バス3を介してトレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4が“1”インクリメントされ(ステップST7)、再度ステップST2に戻り、ステップST2以降の動作が開始される。またステップST2において後方分岐が検出された場合には、ループ認識部10-1のループ終了レジスタ102-1の値が、分岐先レジスタ8の値以下であるか、または分岐元レジスタ9の値以上であるかが判定される(ステップST3)。ステップST3において、肯定(YES)であると判定される場合には、ループ認識部10-1のループ開始レジスタ101-1の値が分岐先レジスタ8の値に等しく、且つループ終了レジスタ102-1の値が分岐元レジスタ9の値に等しいか否かが判定される(ステップST4)。ステップST4において、否定(NO)であると判定される場合には、ループ認識部10-1のループ開始レジスタ101-1には分岐先レジスタ8の値が設定され、ループ終了レジスタ102-1には分岐元レジスタ9の値が設定され、ループトレース・カウンタ103-1にはトレースカウンタ4の値が設定されて、ループ認識部10-1に保持されているループ認識情報が更新され(ステップST6)、前記ステップST7に戻る。ステップST7においては、上述したように、トレース情報バス3を介してトレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。また、前記ステップST4において肯定(YES)であると判定される場合には、トレースカウンタ4に、ループ認識部10-1のループトレース・カウンタ103-1の値が設定されて(ステップST5)、その後に前記ステップST7に戻る。ステップST7においては、トレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。

【0006】一方、ステップST3において否定(NO)であると判定される場合には、ループ認識部10-2のループ終了レジスタ102-2の値が分岐先レジスタ8の値以下であるか、または分岐元レジスタ9の値以上であるかが判定され(ステップST10)、肯定(YES)である

と判定される場合には、ループ認識部10-2のループ開始レジスタ102-2の値が分岐先レジスタ8の値に等しく、且つループ開始レジスタ102-2の値が分岐元レジスタ9の値に等しいか否かが判定される(ステップST11)。ステップST11において否定(NO)と判定される場合には、ループ認識部10-2のループ開始レジスタ101-2には分岐先レジスタ8の値が設定され、ループ終了レジスタ103-2には分岐元レジスタ9の値が設定され、ループトレース・カウンタ103-2にはトレースカウンタ4の値が設定されて、ループ認識部10-2に保持されているループ認識情報が更新される(ステップST13)。次いで、ループ認識部10-1のループ開始レジスタ101-1とループ終了レジスタ102-1に対して、例えば当該レジスタに設定可能な最大値を設定することによりループ認識部10-1の初期化が行われて(ステップST14)、その後に前記ステップST7に戻る。ステップST7においては、トレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。

【0007】なお、ステップST11において肯定(YES)であると判定される場合には、トレースカウンタ4にはループ認識部10-2のループトレース・カウンタ103-2の値が設定され(ステップST12)、前記ステップST7に戻り、トレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。また、前記ステップST10において否定(NO)であると判定される場合には、ループ認識部10-Nのループ終了レジスタ102-Nの値が分岐先レジスタ8の値以下であるか、または分岐元レジスタ9の値以上であるかが判定されて(ステップST20)、肯定(YES)であると判定される場合には、ループ認識部10-Nのループ開始レジスタ101-Nの値が分岐先レジスタ8の値に等しく、且つループ終了レジスタ102-Nの値が分岐元レジスタ9の値に等しいか否かが判定される(ステップST21)。ステップST21において否定(NO)であると判定される場合には、ループ認識部10-Nのループ開始レジスタ101-Nには分岐先レジスタ8の値が設定され、ループ終了レジスタ102-Nには分岐元レジスタ9の値が設定され、ループトレース・カウンタ103-Nにはトレースカウンタ4の値が設定されて、ループ認識部10-Nに保持されているループ認識情報が更新される(ステップST23)。

【0008】次に、ループ認識部10-1および10-2におけるループ開始レジスタ101-1および101-2と、ループ終了レジスタ102-1および102-2に対して、例えば最大値を設定することにより、ループ認識部10-1および10-2の初期化が行われ(ステップST24)、前記ステップST7に戻り、トレース

情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。また、ステップST21において肯定(YES)であると判定される場合には、トレースカウンタ4にループ認識部10-Nのループトレース・カウンタ103-Nの値が設定されて(ステップST22)、同様に、前記ステップST7に戻り、トレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。このことはステップST20において否定(NO)であると判定される場合においても同様であり、前記ステップST7に戻り、トレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。

【0009】

【発明が解決しようとする課題】上述した従来の情報制御装置及びその制御方法においては、後方分岐検出部において検出された情報によってのみトレースの収集制御が行われているために、マイクロプログラムの繰返し実行部分にサブルーチン呼出しが存在し、そこで呼出される子ルーチンの中に、更に繰返し実行部分が存在する場合には、当該子ルーチンの繰返し実行部分の先頭のマイクロプログラム・アドレスが、子ルーチンを呼出した親ルーチンの繰返し実行部分の最終のマイクロプログラム・アドレス以上か、または子ルーチンの繰返し実行部分の最終のマイクロプログラム・アドレスが、前記親ルーチンの繰返し実行部分の最終のマイクロプログラム・アドレス以下という条件に合致することにより、前記子ルーチンにおいて後方分岐が発生した時点において、子ルーチンのループ認識情報がループ認識部に登録されると同時に、親ルーチンのループ認識情報がループ認識部から破棄されてしまう事態となる。

【0010】この結果、前記子ルーチンの繰返し実行部分は、初回と最終回においてのみトレースされるが、前記親ルーチンの繰返し実行部分については、繰返し回数分の全てがトレースされることになる。このように、従来の情報制御装置及びその制御方法においては、マイクロプログラムの繰返し実行部分にサブルーチン呼出しが存在する場合には、効率のよいトレース情報を収集することができないという欠点がある。

【0011】本発明の目的は、上記の課題を解決して、マイクロプログラム中のサブルーチン制御部分を、サブルーチン呼出し命令およびサブルーチンからの復帰命令の検出により認識し、効率のよいトレース情報の収集を可能とする情報制御装置及びその制御方法を実現することにある。

【0012】

【課題を解決するための手段】第1の発明の情報制御装置は、所定のトレース情報を格納保持するトレースメモ

りと、前記トレース情報を前記トレースメモリに格納するためのトレースメモリ・アドレスを保持し、所定のプログラムの命令が実行される都度、前記トレースメモリに当該トレース情報が格納された後に“1”インクリメントするトレースカウンタと、モード指示信号の検出指示を受けて所定のプログラム・アドレスの前回値と今回値とを比較して後方分岐発生の有無を検出する後方分岐検出部と、当該後方分岐発生時における分岐先プログラム・アドレスの値をラッチする分岐先レジスタと、当該後方分岐発生時における分岐元プログラム・アドレスの値をラッチする分岐元レジスタと、前記プログラムの繰返し実行部分を認識するためのループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタを含むループ認識部とを備え、前記プログラムの繰返し実行部分の初回と最終回のみを前記トレースメモリに格納する情報制御装置において、前記プログラムの実行命令が、サブルーチンを呼出す命令であるか否かを検出するCALL命令検出部と、前記プログラムの実行命令が、サブルーチンから復帰する命令であるか否かを検出するRET命令検出部と、前記ループ認識部内のループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタのそれぞれの値を、それぞれ複数セット格納保持するコールスタックと、前記コールスタックに格納されているループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタのそれぞれの値のセット数を計数して保持するコールスタック・カウンタと、を備えて構成され、前記プログラムの繰返し実行部分にサブルーチン呼出し命令が存在する場合には、繰返し実行部分の初回と最終回のみを前記トレースメモリ内に格納することを特徴としている。

【0013】また、第2の発明の情報制御方法は、所定のトレース情報を格納保持するトレースメモリと、前記トレース情報を前記トレースメモリに格納するためのトレースメモリ・アドレスを保持し、所定のプログラムの命令が実行される都度、前記トレースメモリに当該トレース情報が格納された後に“1”インクリメントするトレースカウンタと、モード指示信号の検出指示を受けて所定のプログラム・アドレスの前回値と今回値とを比較して後方分岐発生の有無を検出する後方分岐検出部と、当該後方分岐発生時における分岐先プログラム・アドレスの値をラッチする分岐先レジスタと、当該後方分岐発生時における分岐元プログラム・アドレスの値をラッチする分岐元レジスタと、前記プログラムの繰返し実行部分を認識するためのループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタを含むループ認識部とを備え、前記プログラムの繰返し実行部分の初回と最終回のみを前記トレースメモリに格納する情報制御装置の制御方法において、トレースの開始時に、前記ループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタをそれぞれ初期化する第1のステップ

と、トレース開始後の前記後方分岐検出部による後方分岐検出時に、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、ならびに前記分岐元レジスタの内容と前記ループ終了レジスタの内容とを、それぞれ比較照合する第2のステップと、前記第2のステップにおいて、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、ならびに前記分岐元レジスタの内容と前記ループ終了レジスタの内容とが共に一致した場合に、前記トレース・カウンタに前記ループトレース・カウンタの値を設定する第3のステップと、前記第2のステップにおいて、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、或は前記分岐元レジスタの内容と前記ループ終了レジスタの内容とが不一致の場合に、前記ループ開始レジスタには前記分岐先レジスタの値を、前記ループ終了レジスタには前記分岐元レジスタの値を、前記ループトレース・カウンタには前記トレース・カウンタの値をそれぞれ設定する第4のステップと、前記CALL命令検出部により、サブルーチンと呼出す命令の実行が検出された時に、前記ループ開始レジスタ、前記ループ終了レジスタおよび前記ループトレース・カウンタのそれぞれの値を、前記コールスタック内の前記コールスタック・カウンタの値により指定される位置に保存する第5のステップと、前記コールスタック・カウンタの値を“1”だけ増加させる第6のステップと、前記開始レジスタ、前記ループ終了レジスタおよび前記ループトレース・カウンタをそれぞれ初期化する第7のステップと、前記RET命令検出部により、サブルーチンから復帰する命令の実行が検出された時に、前記コールスタック・カウンタの値を“1”だけ減少させる第8のステップと、前記コールスタック内の前記コールスタック・カウンタの値により指定される位置に保存されている値を、前記ループ開始レジスタ、前記ループ終了レジスタおよび前記ループトレース・カウンタにそれぞれ設定する第9のステップと、を有することを特徴としている。

【0014】更に、第3の発明の情報制御方法は、所定のトレース情報を格納保持するトレースメモリと、前記トレース情報を前記トレースメモリに格納するためのトレースメモリ・アドレスを保持し、所定のプログラムの命令が実行される都度、前記トレースメモリに当該トレース情報が格納された後に“1”インクリメントするトレースカウンタと、モード指示信号の検出指示を受けて所定のプログラム・アドレスの前回値と今回値とを比較して後方分岐発生の有無を検出する後方分岐検出部と、当該後方分岐発生時における分岐先プログラム・アドレスの値をラッチする分岐先レジスタと、当該後方分岐発生時における分岐元プログラム・アドレスの値をラッチする分岐元レジスタと、前記プログラムの繰返し実行部分を認識するためのループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタを含むループ認識部とを備え、前記プログラムの繰返し実行部分の初回

と最終回のみを前記トレースメモリに格納する情報制御装置の制御方法において、トレースの開始時に、前記ループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタをそれぞれ初期化する第1のステップと、トレース開始後の前記後方分岐検出部による後方分岐検出時に、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、ならびに前記分岐元レジスタの内容と前記ループ終了レジスタの内容とを、それぞれ比較照合する第2のステップと、前記第2のステップにおいて、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、ならびに前記分岐元レジスタの内容と前記ループ終了レジスタの内容とが共に一致した場合に、前記トレース・カウンタに前記ループトレース・カウンタの値を設定する第3のステップと、前記第2のステップにおいて、前記分岐先レジスタの内容と前記ループ開始レジスタの内容、或は前記分岐元レジスタの内容と前記ループ終了レジスタの内容とが不一致の場合に、前記ループ開始レジスタには前記分岐先レジスタの値を、前記ループ終了レジスタには前記分岐元レジスタの値を、前記ループトレース・カウンタには前記トレース・カウンタの値をそれぞれ設定する第4のステップと、前記CALL命令検出部により、サブルーチンと呼出す命令の実行が検出された時に、前記コールスタック・カウンタの値を、前記ループ開始レジスタ、前記ループ終了レジスタおよび前記ループトレース・カウンタのそれぞれのサイズの合計サイズ分増加させる第5のステップと、前記RET命令検出部により、サブルーチンから復帰する命令の実行が検出された時に、前記コールスタック・カウンタの値を、前記ループ開始レジスタ、前記ループ終了レジスタおよび前記ループトレース・カウンタのそれぞれのサイズの合計サイズ分減少させる第6のステップと、を有することを特徴としている。

【0015】

【発明の実施の形態】次に、本発明について図面を参照して説明する。

【0016】図1は本発明の1実施形態の構成を示すブロック図である。図1に示されるように、本実施形態は、トレース情報を媒介伝達するトレース情報バス3に対応して、トレース情報を格納するトレースメモリ1と、トレース情報バス3を介して入力されるトレース情報をトレースメモリ1に格納するためのトレースメモリ・アドレスを保持するとともに、当該トレース情報がトレースメモリ1に格納された後に、トレース制御信号2により“1”だけインクリメントされるトレースカウンタ4と、マイクロプログラム・アドレス5の前回値を格納する前回値レジスタ71およびマイクロプログラム・アドレス5の今回値を格納する今回値レジスタ72を含み、モード指示信号6により検出指示される際に、マイクロプログラム・アドレス5の前回値と今回値とを比較して、前回値が今回値以上である場合に、後方分岐が発

11

生したことを検出する後方分岐検出部7と、当該後方分岐検出部7において後方分岐が検出された際の前回値レジスタ71を保持する分岐先レジスタ8と、後方分岐検出部7において後方分岐が検出された際の今回値レジスタ72を保持する分岐元レジスタ9と、それぞれループ開始レジスタ101-i (i=1, 2, ……………, N)、ループ終了レジスタ102-i、およびループトレース・カウンタ103-i (i=1, 2, ……………, N)を含み、マイクロプログラムの繰返し実行部分を認識するための情報を保持するループ認識部10-1、10-2、……………、10-Nと、マイクロプログラムの実行命令が、サブルーチンを呼出す命令であるか否かを検出するCALL命令検出部21と、マイクロプログラムの実行命令が、サブルーチンから復帰する命令であるか否かを検出するRET命令検出部22と、サブルーチン呼出し時に、ループ認識部10-1、10-2、……………、10-Nに保持されているループ認識情報が保存され、サブルーチンからの復帰時に、これらのループ認識部に保存されているループ認識情報を取出すために使用されるFIFO型の領域であるコールスタック23と、当該コールスタック23に格納されているループ認識情報の数を示すコールスタック・カウンタ24とを備えて構成される。

【0017】なお、従来例の場合と同様に、上記の各ループ認識部内のループ開始レジスタ101-iには、マイクロプログラムの繰返し部分の先頭マイクロプログラム・アドレスが格納され、ループ終了レジスタ102-iには、マイクロプログラムの繰返し部分の最終マイクロプログラム・アドレスが格納され、ループトレース・カウンタ103-iには、マイクロプログラムの繰返し部分認識時のトレースカウンタ4の値が格納される。また、これらのループ認識部に対しては、それぞれに、例えば大小関係による重み付けされたループ識別IDが付与されている。

【0018】また、図2は、本実施形態の動作手順を示すフローチャートである。図4との対比により明らかなように、本実施形態においては、CALL命令検出部21、RET命令検出部22、コールスタック23およびコールスタック・カウンタ24等を含む装置構成の付加に伴ない、処理ステップとしては、ステップST31、ST32、ST33、ST34、ST35およびST36等を含むステップが新たに付加されている。

【0019】次に、図1および図2を参照して、本従来例の動作について説明する。なお、従来例の場合と同様に、説明の都合上、ループ認識部が、図1に記載されているループ認識部10-1、ループ認識部10-2およびループ認識部10-Nの3個のループ認識部のみが設けられている場合を想定して動作説明を行うものとする。

【0020】トレース動作開始時においては、全てのル

12

ープ認識部の内部に含まれるループ開始レジスタおよびループ終了レジスタは初期化される(ステップST1)。次いで、コールスタック・カウンタ24は例えば“0”に初期化され(ステップST31)、トレース動作中に、サブルーチン呼出し命令が実行されたことがCALL命令検出部21において検出されたか否かが判定されて(ステップST32)、サブルーチン呼出し命令の実行が検出される場合には、ループ認識部10-1、10-2および10-Nに保持されている、ループ開始レジスタ101-1、101-2および101-Nの値と、ループ終了レジスタ102-1、102-2および102-Nの値と、ループトレース・カウンタ103-1、103-2および103-Nの値とを含むループ認識情報が、コールスタック・カウンタ24の値により指定されるコールスタック23内の所定位置に保存され、コールスタック・カウンタ24は“1”インクリメントされる(ステップST33)。そして、ループ認識部10-1、10-2および10-N内のループ認識情報は初期化されて(ステップST34)、トレース情報バス3を介してトレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4が“1”インクリメントされ(ステップST7)、再度ステップST2に戻り、ステップST2以降の動作が開始される。また、ステップST32においてサブルーチン呼出し命令の実行が検出されない場合には、トレース動作中に、サブルーチン復帰命令が実行されたことがRET命令検出部22において検出されたか否かが判定されて(ステップST35)、サブルーチン復帰命令の実行が検出される場合には、コールスタック・カウンタ24が“1”デクリメントされ、コールスタック・カウンタ24により指定されるコールスタック23内の所定位置に保存されている値が取出されて、ループ認識部10-1、10-2および10-N内のループ認識情報として設定される(ステップST36)。またステップST35において、サブルーチン復帰命令の実行が検出されない場合には、後方分岐検出部7においては後方分岐が検出されたか否かが判定されて(ステップST2)、検出されていない場合には、トレース情報バス3を介してトレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4が“1”インクリメントされ(ステップST7)、再度ステップST2に戻り、ステップST2以降の動作が開始される。またステップST2において後方分岐が検出された場合には、ループ認識部10-1のループ終了レジスタ102-1の値が、分岐先レジスタ8の値以下であるか、または分岐元レジスタ9の値以上であるかが判定される(ステップST3)。

【0021】ステップST3において、肯定(YES)であると判定される場合には、ループ認識部10-1のループ開始レジスタ101-1の値が分岐先レジスタ8の値に等しく、且つループ終了レジスタ102-1の値が分

13

岐元レジスタ9の値に等しいか否かが判定される(ステップST4)。ステップST4において、否定(NO)であると判定される場合には、ループ認識部10-1のループ開始レジスタ101-1には分岐先レジスタ8の値が設定され、ループ終了レジスタ102-1には分岐元レジスタ9の値が設定され、ループトレース・カウンタ103-1にはトレースカウンタ4の値が設定されて、ループ認識部10-1に保持されているループ認識情報が更新され(ステップST6)、前記ステップST7に戻る。ステップST7においては、上述したように、トレース情報バス3を介してトレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。また、前記ステップST4において肯定(YES)であると判定される場合には、トレースカウンタ4に、ループ認識部10-1のループトレース・カウンタ103-1の値が設定されて(ステップST5)、その後前記ステップST7に戻る。ステップST7においては、トレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。

【0022】一方、ステップST3において否定(NO)であると判定される場合には、ループ認識部10-2のループ終了レジスタ102-2の値が分岐先レジスタ8の値以下であるか、または分岐元レジスタ9の値以上であるかが判定され(ステップST10)、肯定(YES)であると判定される場合には、ループ認識部10-2のループ開始レジスタ102-2の値が分岐先レジスタ8の値に等しく、且つループ開始レジスタ102-2の値が分岐元レジスタ9の値に等しいか否かが判定される(ステップST11)。ステップST11において否定(NO)と判定される場合には、ループ認識部10-2のループ開始レジスタ101-2には分岐先レジスタ8の値が設定され、ループ終了レジスタ103-2には分岐元レジスタ9の値が設定され、ループトレース・カウンタ103-2にはトレースカウンタ4の値が設定されて、ループ認識部10-2に保持されているループ認識情報が更新される(ステップST13)。次いで、ループ認識部10-1のループ開始レジスタ101-1とループ終了レジスタ102-1に対して、例えば当該レジスタに設定可能な最大値を設定することによりループ認識部10-1の初期化が行われて(ステップST14)、その後前記ステップST7に戻る。ステップST7においては、トレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。

【0023】なお、ステップST11において肯定(YES)であると判定される場合には、トレースカウンタ4にはループ認識部10-2のループトレース・カウンタ103-2の値が設定され(ステップST12)、前記ステッ

14

プST7に戻り、トレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。また、前記ステップST10において否定(NO)であると判定される場合には、ループ認識部10-Nのループ終了レジスタ102-Nの値が分岐先レジスタ8の値以下であるか、または分岐元レジスタ9の値以上であるかが判定されて(ステップST20)、肯定(YES)であると判定される場合には、ループ認識部10-Nのループ開始レジスタ101-Nの値が分岐先レジスタ8の値に等しく、且つループ終了レジスタ102-Nの値が分岐元レジスタ9の値に等しいか否かが判定される(ステップST21)。ステップST21において否定(NO)であると判定される場合には、ループ認識部10-Nのループ開始レジスタ101-Nには分岐先レジスタ8の値が設定され、ループ終了レジスタ102-Nには分岐元レジスタ9の値が設定され、ループトレース・カウンタ103-Nにはトレースカウンタ4の値が設定されて、ループ認識部10-Nに保持されているループ認識情報が更新される(ステップST23)。

【0024】次に、ループ認識部10-1および10-2におけるループ開始レジスタ101-1および101-2と、ループ終了レジスタ102-1および102-2に対して、例えば最大値を設定することにより、ループ認識部10-1および10-2の初期化が行われ(ステップST24)、前記ステップST7に戻り、トレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。また、ステップST21において肯定(YES)であると判定される場合には、トレースカウンタ4にループ認識部10-Nのループトレース・カウンタ103-Nの値が設定されて(ステップST22)、同様に、前記ステップST7に戻り、トレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。このことはステップST20において否定(NO)であると判定される場合においても同様であり、前記ステップST7に戻り、トレース情報をトレースメモリ1に格納するとともに、トレースカウンタ4を“1”インクリメントして、再度ステップST2以降の処理が行われる。

【0025】以上のようにして、サブルーチンを呼出した親ルーチンと呼出された子ルーチンとの間において、ループ認識部において保持されているループ認識情報を切替えることができるために、サブルーチンを含むマイクロプログラムにおいても、効率のよいトレース動作を行うことが可能となる。即ち、本発明によれば、マイクロプログラム中のサブルーチン制御部分を、サブルーチン呼出し命令およびサブルーチンからの復帰命令の検出により認識し、効率のよいトレース情報の収集を可能とす

る情報制御装置およびその制御方法が実現される。

【0026】なお、上記の説明においては、コールスタック・カウンタが、コールスタックに格納されているループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタのそれぞれの値のセット数を計数して保持しており、トレース動作中に、サブルーチン呼出し命令が実行されたことがCALL命令検出部において検出された際には、全ループ認識部に保持されているループ認識情報がコールスタックに格納された後に、コールスタック・カウンタを“1”インクリメントし、またトレース動作中に、サブルーチン復帰命令が実行されたことがRET命令検出部において検出された際には、コールスタック・カウンタを“1”デクリメントした後に、コールスタックに格納されているループ認識情報を取出して、全ループ認識部に設定するという情報制御方法について説明したが、この制御方法とは別に、コールスタックがメモリで実装されており、コールスタック・カウンタが、前記コールスタックのアドレスを保持している場合に、トレース動作中に、サブルーチン呼出し命令が実行されたことがCALL命令検出部において検出された際には、全ループ認識部において保持されているループ認識情報をコールスタックに格納した後に、コールスタック・カウンタの値をループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタのそれぞれのサイズの合計サイズ分増加させ、またトレース動作中に、サブルーチン復帰命令が実行されたことがRET命令検出部において検出された際には、コールスタック・カウンタの値を、ループ開始レジスタ、ループ終了レジスタおよびループトレース・カウンタのそれぞれのサイズの合計サイズ分減少した後に、コールスタックに格納されているループ認識情報を取出して、全ループ認識部に設定するという情報制御方法を用いてもよい。また、以下においては、具体的にマイクロプログラムの1例の適用例を参照して、本発明の効果がどのようなものであるかについて説明する。

【0027】図5(a)は、サブルーチンを含んだマイクロプログラムの1例である。また図5(b)は、従来の情報制御装置により、上記の図5(a)に示されるマイクロプログラムをトレースした結果を示す図であり、図5(c)は、本発明により、図5(a)に示されるマイクロプログラムをトレースした結果を示す図である。従来の情報制御装置及びその制御方法においては、図5(b)に示されるように、親ルーチンの処理Aおよび処理Bの繰返し処理の2回目から(N-1)回目のトレース情報およびその時のルーチンの処理Cの繰返し処理の初回と最終回のトレース情報が、トレースメモリ内に残留しているのに対して、本発明においては、図5(c)に示されるように、前記トレース情報がトレースメモリ内に残留しないように改善されている。図5(a)に示されるマイクロプログラクの処理A、処理Bおよび処理

Cにおけるトレースメモリ消費量を共に500バイト、繰返し数NおよびMを共に10と仮定すると、従来の情報制御装置およびその方法におけるトレースメモリの消費量が16000バイトであるのに対して、本発明におけるトレースメモリの消費量は4000バイトに止まり、トレースメモリの利用効率が4倍に改善される。従って、同一容量のトレースメモリにより、4倍のマイクロプログラムの実行結果を当該トレースメモリ内に残すことができ、トレースデータによるマイクロプログラムのデバッグ可能な範囲が4倍に拡大され、デバッグ効率を向上させることができる。

【0028】

【発明の効果】以上説明したように、本発明は、CALL命令検出部およびRET命令検出部を備えることにより、マイクロプログラムのサブルーチン呼出し命令の実行および復帰命令の実行を検出し、当該サブルーチン命令の検出時においては、検出時点まで使用されていたループ認識部のループ認識情報をコールスタックに保持しておき、サブルーチン復帰命令検出時においては、前回サブルーチン呼出しによりコールスタックに保持されていたループ認識情報をループ認識部に戻すようにして、親ルーチンと子ルーチンとの間において、ループ認識部に保持されているループ認識情報を切替えてトレース動作を行うようにしたことにより、マイクロプログラムの繰返し実行部分にサブルーチン呼出しが含まれている場合においても、繰返し実行部分の初回と最終回に対してのみトレース情報が残されることになり、サブルーチンを含んだマイクロプログラムのトレースにおいて、同一容量のトレースメモリにより、より一層広範囲のマイクロプログラムの実行結果をトレースメモリに残すことが可能となり、これにより、トレースデータによるマイクロプログラムのデバッグ可能範囲が拡大され、デバッグ効率を向上させることができるという効果がある。

【図面の簡単な説明】

【図1】本発明の1実施形態の構成を示すブロック図である。

【図2】前記1実施形態における動作手順のフローチャートを示す図である。

【図3】従来例の構成を示すブロック図である。

【図4】前記従来例における動作手順のフローチャートを示す図である。

【図5】マイクロプログラムおよびマイクロプログラムのトレース・データを示す図である。

【符号の説明】

- | | |
|---|----------------|
| 1 | トレースカウンタ |
| 2 | トレース制御信号 |
| 3 | トレース情報バス |
| 4 | トレース・カウンタ |
| 5 | マイクロプログラム・アドレス |
| 6 | モード指示信号 |

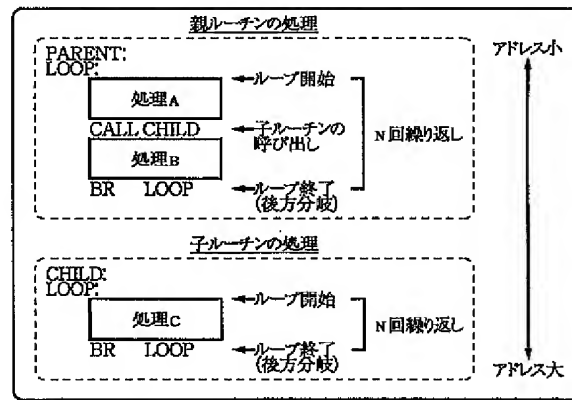
17

18

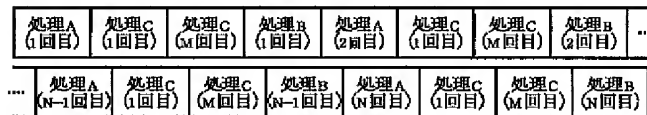
7 後方分岐検出部
 8 分岐先レジスタ
 9 分岐元レジスタ
 10-1、10-2、……、10-N ループ認識部
 21 CALL命令検出部
 22 RET命令検出部
 23 コールスタック
 24 コールスタック・カウンタ

101-1、101-2、……、101-N ループ
 開始レジスタ
 102-1、102-2、……、102-N ループ
 終了レジスタ
 103-1、103-2、……、103-N ループ
 トレース・カウンタ
 ST1～ST24、ST31～ST36 ステップ

【図5】



(a)

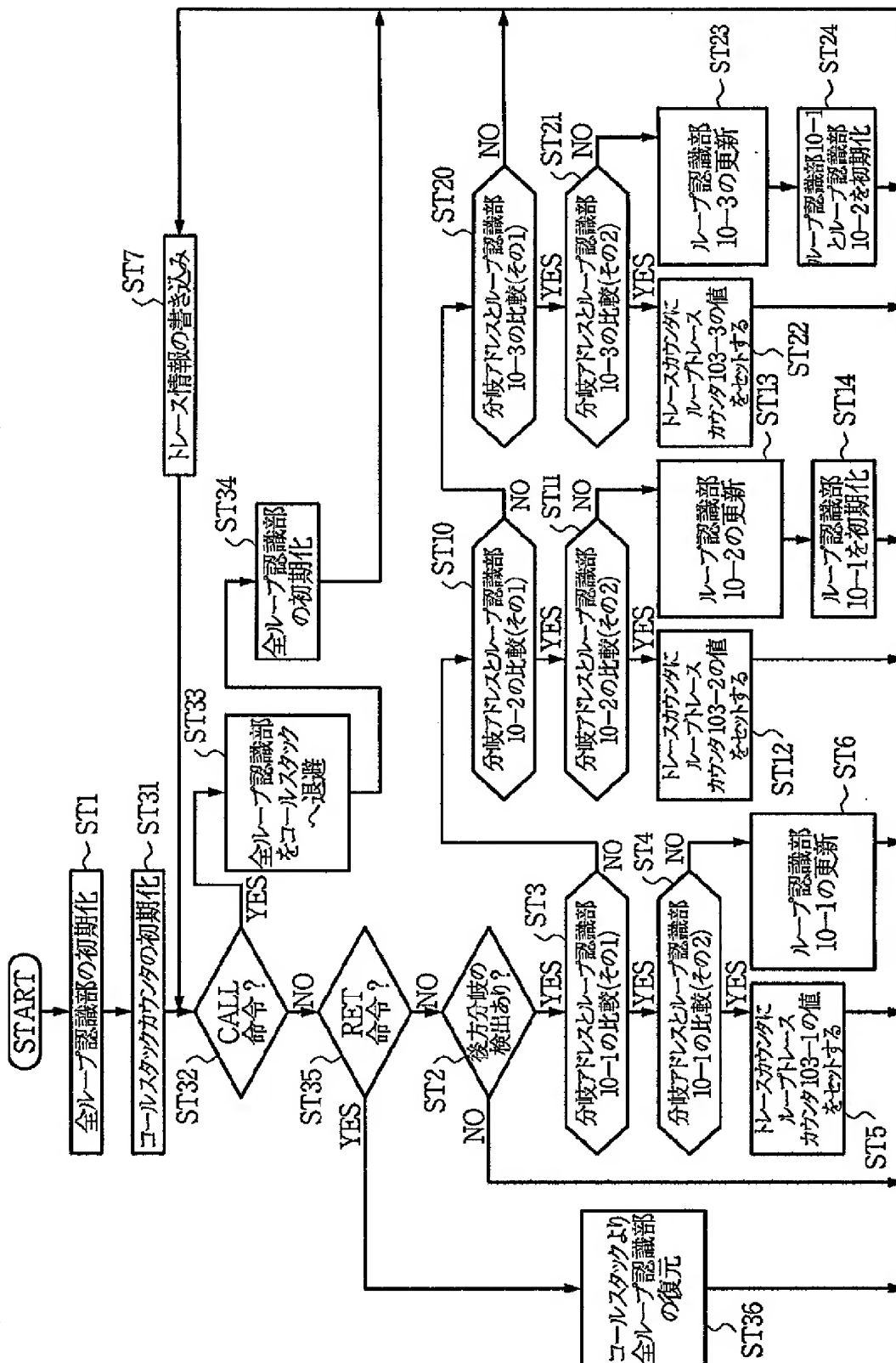


(b)



(c)

【図 2】



【図3】

